R. E. Smith*
NASA Langley Research Center
Hampton, Virginia

Abstract

An algebraic procedure for the generation of boundary-fitted grids about wing-fuselage configurations is presented. A wing-fuselage configuration is specified by cross sections and mathematically represented by Coons' patches. A configuration is divided into sections so that several grid blocks that either adjoin each other or partially overlap each other can be generated. Each grid has six exterior surfaces that map into a computational cube. Grids are first determined on the six boundary surfaces and then in the interior. Grid curves that are on the surface of the configuration are derived from the intersection of planes with the Coons' patch definition. Single-valued functions relating approximate arc lengths along the grid curves to a computational coordinate define the distribution of grid points. The two-boundary technique and transfinite interpolation are used to determine the boundary surface grids that are not on the configuration, and transfinite interpolation with linear blending functions is used to determine the interior grid.

Introduction

Grid generation is an essential part of the numerical solution of systems of partial differential equations.[1,2] In computational fluid dynamics, grid generation is primarily associated with the application of finite difference and finite volume methods to obtain numerical solutions of various subsets of the Navier-Stokes equations which will be henceforth called the governing equations.[3,4] The governing equations are defined in a physical coordinate system and are then transformed to an idealized rectangular computational coordinate system (computational cube). Additional unknowns called transformation derivatives, which are the elements of the Jacobian matrix, are introduced into the transformed governing equations, and the computational coordinates become new independent variables.

An approach for the discrete determination of the transformation derivatives is to superimpose a boundary-fitted grid onto the physical domain which corresponds to a uniform grid on the computational domain. Numerically differentiating the physical grid with respect to the computational grid and applying matrix operations determine the transformation derivatives.[3] The process of determining a "one-to-one" relationship between the physical grid and the computational grid from which the transformation derivatives are computed is grid generation. The advantages of using transformed governing equations and grid generation are: (1) Reduced complexity and added accuracy in the application of boundary conditions, and (2) the ability to increase solution accuracy in regions with high gradients through grid clustering.

*Senior Research Engineer

For creating grids about wing-fuselage bodies, an algebraic approach based on Coons' patch-surface definition,[5] the two-boundary grid generation technique,[3,4,6] and transfinite interpolation[6] is presented. The process is divided into five steps:

1. Define the wing and fuselage surfaces and their intersection (Fig. 1).

2. Determine grids on the wing and fuselage surfaces which will map to adjoining exterior sides of a computational cube (Fig. 2).

3. Define outer physical boundaries (grid curves) corresponding to edges of the computational cube (Fig. 3).

4. Determine grids on the exterior surfaces other than the wing and fuselage which correspond to the remaining exterior sides of the computational cube (Fig. 4).

5. Determine the interior grid using transfinite interpolation with linear blending functions (Fig. 5).

The primary concepts that are described herein have been developed in references 7 and 8. They are amplified and extended in this paper. The approach combines aircraft-surface definition concepts that have been developed over many years and algebraic grid generation concepts. For complex aircraft configurations, it is recognized that it is unlikely that a single grid that maps into a computational cube can be devised. It will probably require several grids that either adjoin each other or partially overlap each other. For the wing-fuselage grids described herein, four adjoining grids are specified. They are the top-front, bottom-front, top-back and bottom-back grids, and most of the step-by-step demonstration of the grid generation procedure will be with the top-front grid.

Wing-Fuselage Surface Definition

The components of an aircraft can be described by the Harris geometry[9] in terms of cross sections. A fuselage is described by cross sections along the x-body axis, and a wing is described as airfoil sections in the spanwise z-direction (Fig. 6). In turn, each airfoil section is defined by the coordinates of the camber line and Δy coordinates. In reference 10 cubic splines are fit along and across the specified cross-sectional data for each component. The specified positional data and derivative data obtained from the spline fits provide corner parameters for a Coons' patch-surface definition for the component. Each patch is of the form

$$V(u,w) = UCBC^{T}W^{T} \qquad (1)$$

where

$$V(u,w) = [x(u,w) \quad y(u,w) \quad z(u,w)]$$

$$B = \begin{bmatrix} V(0,0) & V(0,1) & V_w(0,0) & V_w(0,1) \\ V(1,0) & V(1,1) & V_w(1,0) & V_w(1,1) \\ V_u(0,0) & V_u(0,1) & V_{uw}(0,0) & V_{uw}(0,1) \\ V_u(1,0) & V_u(1,1) & V_{uw}(1,0) & V_{uw}(1,1) \end{bmatrix}$$

$$C = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$U = [u^3 \quad u^2 \quad u \quad 1], \quad W = [w^3 \quad w^2 \quad w \quad 1],$$

$$0 \leq u \leq 1, \qquad 0 \leq w \leq 1$$

The vector-valued function V(u,w) represents the Cartesian coordinates (x,y,z) for a patch definition, and the variables u and w are independent parametric variables. The limits of the parametric variables are zero and one and define the corners of a patch (Fig. 7). The subscripts u and w in matrix B denote differentiation with respect to the parametric independent variables. Using cubic spline functions to compute the corner derivatives (only the patch corner positions are pre-specified) and setting the cross derivatives equal to zero (lower right elements of the matrix B), the patch description is a bi-cubic representation (zero cross derivatives) of the region bounded by the sides of the patch. An ordered set of patches represent the surface of a component, and reference 10 describes a widely available computer program for performing the splines fits, saving the patch description and plotting enriched surface components (Fig. 1).

Wing-Fuselage Intersection

In order to map the wing and fuselage surfaces into two adjoining planar surfaces in the computational coordinate system, a method of determining the intersection of the two surfaces in the physical coordinate system must be found. Since the two components are described mathematically by sets of patch equations, it is theoretically possible to solve the two sets of equations simultaneously for the intersection; however, this is not the approach that is taken in this paper. Instead, the mathematical process of finding the intersection of a sequence of planes with the wing and fuselage components, and then computing the intersection of the planar curves is used.[7] The reason for taking this approach is that the intersection of arbitrarily defined planes (three unique points) with the components of a configuration (accumulation of patch equations in the intersection region for each component) is a facility of the surface-definition program described in reference 10. The plane-patch intersection is obtained by rewriting equation (1) as

$$x = UCB(x)C^TW^T$$

$$y = UCB(y)C^TW^T$$

$$z = UCB(z)C^TW^T$$

and substituting the components in the equation of a plane

$$ax + by + cz + d = 0$$

where a, b, c, and d are constants derived from a three point description. This leads to the equation

$$UGW^T + d = 0 \qquad (2)$$

where

$$G = aCB(x)C^T + bCB(y)C^T + cCB(z)C^T$$

is a matrix of constants. Assigning a value to one of the parametric variables u or w results in equation (2) becoming a cubic polynomial with respect to the other parametric variable. Finding the real root of equation (2) interior to the patch boundaries for a sequence of the specified parametric variable defines the curve of the intersection of a patch and a plane (Fig. 8).

The intersection of a wing-surface component and a fuselage-surface component is found by first determining the intersection of the patch definition of each component with a sequence of planes along and orthogonal to the x-body axis (Fig. 9). This is accomplished by using the computer code described in reference 10 as a kernel that is driven by another code which specifies the planes, orders the coordinates, deletes multiple points at the intersection of neighboring patches, and searches to find the intersection of the planar curves from the two components (Fig. 10). The leading point and trailing point of the wing-fuselage intersection occur where there is only one intersection point on the plane-patch intersection curves. A search is performed by the driver code to precisely define these terminating points.

Coordinate Axes, Grids and Indices

For the grids described herein, the physical point corresponding to the origin of the computational coordinate system is the leading point of the wing-fuselage intersection, and the wing-fuselage intersection curve corresponds to the ξ-axis. When back grids are considered, the ξ-axis extends down the fuselage behind the wing. A curve that will be described at a later point that spans

from the leading wing-fuselage intersection point to the leading nose point on the fuselage will correspond to the η-axis, and a curve along the leading edge of the wing from the wing-fuselage intersection point to the wing tip will correspond to the ζ-axis (Fig. 11). A grid is represented by $F_i(I,J,K)$ where

$$F_i(I,J,K) = [X_i(I,J,K) \quad Y_i(I,J,K) \quad Z_i(I,J,K)]$$

and $I = 1,2,\ldots L$, $J = 1,2,\ldots M$, and $K = 1,2,\ldots N$. The top-front fuselage grid is denoted by $F_1(I,J,K)$. The computational coordinates are related to indices I, J, and K by

$$\xi = (I - 1)/(L - 1),$$

$$\eta = (J - 1)/(M - 1)$$

$$\zeta = (K - 1)/(N - 1)$$

where $I = 1,2,\ldots L$, $J = 1,2,\ldots M$, and $K = 1,2,\ldots N$.

Fuselage Grid

The fuselage grid computation is preceded by the computation of a distribution of grid points on the wing-fuselage intersection curve ($F_1(I,1,1)$, $I = 1,2,\ldots L$) and along a curve on top of the fuselage at $\eta = 1$ ($F_1(I,M,1)$, $I = 1,2,\ldots L$). Note that the configuration is symmetric about the x-y plane at $z = 0$ (Fig. 1). The computational ξ-coordinate is uniformly discretized on the unit interval $0 \leq \xi \leq 1$, and the ξ-coordinate is related by "one-to-one" functions to the normalized-approximate arc length along the wing-fuselage intersection curve and along the top-boundary curve. Considering only the top-front grid, the normalization factors for the arc lengths are the approximate arc-length value at the trailing wing-fuselage intersection point and the approximate arc-length value on the top-boundary curve at the terminal point which has the same x-coordinate as the trailing wing-fuselage intersection point (Fig. 12). The physical coordinates of the intersection curve and the top-boundary curve are related to the normalized approximate arc lengths by interpolation into previously stored tables of coordinates versus arc lengths. In accordance with the techniques described in references 6 and 11, grid spacing in the physical domain is controlled by single-valued functions relating computational coordinates to normalized approximate arc lengths. Low slopes in these functions correspond to a physical grid concentration and high slopes correspond to physical-grid dispersions (Fig. 13).

With the distribution of grid points along the wing-fuselage intersection curve and the fuselage-top boundary curve, dense sets of points are computed on fuselage surface curves and stored. This is done with the Coons' patch-surface representation of the fuselage and the plane-patch intersection capability previously described. In this case the planes are defined by three points of which two are on the boundary curves $\eta = 0$ and $\eta = 1$. The third point is defined to be on the x-body axis with the x-coordinate equal to the x-coordinate of the wing-fuselage grid point (Fig. 14). Using the computer program described in reference 10 as a kernel, a driver program specifies the points defining the plane, orders the plane-patch intersection points, deletes multiple points, computes the approximate arc length along the curves from the wing-fuselage intersection boundary to the fuselage-top boundary, and stores the coordinates of the curves as a function of the normalized approximate arc length. The distribution of the grid points in the η-direction is obtained in the same manner as previously discussed for the wing-fuselage intersection and the top boundary. Single-valued functions between the η-coordinate and the normalized approximate arc lengths are formed where low slopes result in grid concentration and high slopes result in grid dispersion. Figure 15 shows a fuselage grid with concentrations toward the leading and trailing wing-fuselage intersection points and toward the wing-fuselage intersection curve. The fuselage surface grid is represented by $F_1(I,J,1)$, $I = 1,2,\ldots L$, $J = 1,2,\ldots M$.

Wing Grid

The plane-patch intersection capability and arc-length spacing-control functions are again used to determine the wing grid. Also, the concept of transfinite interpolation[6] is used to conform the grid points near the wing-fuselage intersection to the shape of the wing-fuselage intersection curve. The process is to first establish the normalized approximate arc lengths along the leading and trailing edges of the wing from the leading and trailing wing-fuselage intersection points to the wing tip and to express the coordinates as tabular functions of the arc lengths. In accordance with the technique described above and in reference 11, a spacing-control function relating the ζ-coordinate to the normalized approximate arc length is determined. In this paper, the control functions are either uniform or increase exponentially in the direction of the wing tip, and the same function is used for the leading edge and the trailing edge. Given the spacing distributions and a discrete set of ζ-coordinates, physical coordinates $F_1(1,1,K)$, $K = 1,2,\ldots N$, $F_1(L,1,K)$, $K = 1,2,\ldots N$) are computed on the leading edge and the trailing edge of the wing. For each ζ-coordinate, a plane-patch intersection on the wing is computed with the third point for the plane specified by the X- and Z-coordinates of the the leading-edge point and a distinctly different Y-coordinate. Again, the computer program described in reference 10 is used as a kernel. A driver code defines the planes, orders the points from leading edge to trailing edge, deletes multiple points, and sets up tables of coordinates as a function of normalized approximate arc length. The grid-spacing distribution function for the wing-fuselage intersection curve in the ξ-coordinate direction is used to compute an intermediate wing-surface grid $E(I,K)$, $I = 1,2,\ldots L$, $K = 1,2,\ldots N$. In order to conform the

wing-surface grid to the shape of the wing-fuselage intersection curve the final wing-surface grid is computed by

$$F_1(I,1,K) = E(I,K) + [1 - \zeta(K)][F_1(I,1,1) - E(I,1)] \quad (3)$$

where

$$\zeta(K) = \frac{e^{-k[(K-1)/(N-1)]} - 1}{e^{-k} - 1}$$

$I = 1,2,...L$, $K = 1,2,...N$, and k is positive constant. Figure 16 shows a wing-fuselage surface grid with grid concentration toward the wing-fuselage intersection curve.

Unconstrained Surface Grids

Six physical grid surfaces from the top-front grid correspond to the six sides of the computational cube. Two of the surfaces are constrained to lie on the prespecified wing and fuselage surfaces, and they have been discussed above. The remaining four grid surfaces are not constrained to prespecified surfaces, and they are referred to as the side, front, back, and cap grid surfaces. The side grid surface can be defined in the x-y plane down the center of the fuselage. The front, back, and cap grid surfaces conform to the wing, fuselage, and side surfaces. Since this grid is planar and the bottom grid boundary is specified from the fuselage grid, a two-dimensional version of the two-boundary technique as described in references 3 and 11 can be applied directly with minor changes to preexisting software. Also, other two-dimensional techniques and software could be used to generate this grid. Using the two-boundary technique, only the outer-boundary grid curve ($F_1(I,M,N)$, $I = 1,2,...L$) and the clustering control for the cubic-connecting function must be user specified. The top-front side grid surface is denoted by $F_1(I,M,K)$, $I = 1,2,...L$, $K = 1,2,...N$.

Defining the side-grid surface specifies the grid-point distributions for the adjoining-boundary grid curves for the front and back grid surfaces (Fig. 17). Also, the grid distributions along the boundary grid curves for the fuselage and wing specify two more boundary grid-curve distributions for the front and back grid surfaces (Fig. 17). In this paper, the outer boundary grid curves ($F_1(I,J,N)$, $J = 1,2,...M$, and $F_1(L,J,N)$, $J = 1,2,...M$) are specified to be planar, and the distribution of grid points along these curves is based on normalized approximate arc-length distributions as previously discussed. The front and back grid surfaces, however, are not planar since the wing-boundary grid curves are not planar. A three-dimensional version of the two-boundary technique is incorporated into the concept of transfinite interpolation to compute the front and back grid surfaces. The two-boundary technique is used to compute intermediate grid surfaces ($E(1,J,K)$ and $E(L,J,K)$, $J = 1,2,...M$, $K = 1,2,...N$) connecting the fuselage-boundary grid curves to the outer-boundary grid curves. The only additional information that must be specified is a control function which defines the grid clustering along the cubic-connecting function. Transfinite interpolation is used in a similar manner as in the wing-grid definition. In this instance, it is used to conform the front and back grid surfaces to the side-boundary grid curve and the wing-boundary grid curve. The formulation is

$$F_1(I,J,K) = E(I,J,K) + \alpha(J)[F_1(I,1,K) - E(I,1,K)] + \beta(J)[F_1(I,M,K) - E(I,M,K)] \quad (4)$$

where

$$\alpha(J) = 1 - \frac{e^{-k[(J-1)/M-1)]} - 1}{e^{-k} - 1}$$

$$\beta(J) = \frac{e^{k[(J-1)/(M-1)]} - 1}{e^{k} - 1}$$

and $I = 1$ and L. The parameter k controls the effect of the side boundaries at $J = 1$ and M into the interior grid $J = 2,3...M - 1$.

All the boundary grid curves for the cap grid surface are at this point defined (i.e., $F_1(I,J,N)$, $I = 1$ and L, $J = 1,2...M$, $F_1(I,J,N)$, $I = 1,2...L$, $J = 1$ and M (Fig. 18)). It would initially appear that a two-step transfinite-interpolation formulation similar to equation (4) could be used to generate the cap grid surface. The formulation is

$$E(I,J) = \alpha(I)\, F_1(1,J,N) + \beta(I)\, F_1(L,J,N) \quad (5)$$

where

$$\alpha(I) = 1 - \frac{e^{-k[(I-1)/(L-1)]} - 1}{e^{-k} - 1}$$

$$\beta(I) = \frac{e^{k[(I-1)/(L-1)]} - 1}{e^{k} - 1}$$

and

$$F_1(I,J,L) = E(I,J) + \sigma(J)\,[F_1(I,1,N) - E(I,1)] + \mu(J)\,[F_1(I,M,N) - E(I,M)] \quad (6)$$

where

$$\sigma(J) = 1 - \frac{e^{-\bar{k}[(J-1)/(M-1)]} - 1}{e^{\bar{k}} - 1}$$

and

$$\mu(J) = \frac{e^{\bar{k}[(J-1)/(M-1)]} - 1}{e^{\bar{k}} - 1}$$

The magnitude of the parameters k and $\bar{k}$ extends the effect of the physical-boundary grid curves into the interior of the grid surface. Unfortunately, the generation of the cap grid surface using this approach results in an unwanted indentation of the surface. Possibly there exist blending functions that would solve this problem, but they have not been found. A second approach that solves the indentation problem, but is not entirely satisfactory, is the specification of more points on the cap surface and applying bi-directional tension-spline functions -- first in the η-direction and then in the ξ-direction. Approximate arc length is again used as an intermediate variable, and the resulting cap grid surface ($F_1(I,J,N)$, $I = 1,2,...L$, $J = 1,2,...M$) is smooth and has no undesirable indentation (Fig. 19).

The reason that the second approach is not entirely satisfactory is that there is no consistent way of identifying where the additional specified points should be located on the surface. A third approach is to specify the cap surface by a Coons' patch definition similar to the fuselage definition. That is, the cap surface could be specified by cross-sections along the x-axis. Grid curves between the wing-tip boundary grid curve and the top-side boundary grid curve could be found by using the plane-patch intersection capability previously described for the fuselage. The curves would be stored as a function of normalized approximate arc length and the grid-point distributions along these curves could be derived by linear blending with the distributions of grid points along the front-boundary grid curve and the back-boundary grid curve.

Interior Grid Computation

Once the exterior grid surfaces corresponding to the six sides of the computational cube have been found, an interior grid is described by transfinite interpolation using linear-blending functions. That is:

$$D(I,J,K) = [1 - (K-1)/(N-1)][F_1(I,J,1)] + [(K-1)/(N-1)][F_1(I,J,N)] \quad (7)$$

$$E(I,J,K) = D(I,J,K) + [1 - (J-1)/(M-1)][F_1(I,1,K) - D(I,1,K)] + [(J-1)/(M-1)][F_1(I,M,K) - D(I,M,K)] \quad (8)$$

$$F_1(I,J,K) = E(I,J,K) + [1 - (I-1)/(L-1)][F_1(1,J,K) - E(1,J,K)] + [(I-1)/(L-1)][F_1(1,J,K) - E(L,J,K)] \quad (9)$$

where $I = 1,2,...L$, $J = 1,2,...M$, and $K = 1,2,...N$. Selected interior-grid surfaces for the top-front grid are shown in Figure 5. Selected interior-grid surfaces for the top-front and bottom-front grids are shown in Figure 20. Figures 21 and 22 show the back-top and bottom grids. Grid concentrations in the latter figures are at the leading and trailing edges of the wing and toward the wing-fuselage intersection.

Other Configurations

There are many topological complexities that can occur on real wing-fuselage configurations that have not been considered in this paper. For instance, placing the wing very high or very low relative to the fuselage creates too much grid concentration on the part of the fuselage near the wing-fuselage intersection. Also, if the wing has a sharp leading edge, the C-type of grid may not be satisfactory. On the other hand, the component parts of the procedure such as surface representation, patch-plane intersections and the grid spacing control are applicable to a wide variety of grid generation problems. Two additional examples with different topological considerations are shown here. The first example is a canard-fuselage grid (Fig. 23) where the canard has a sharp leading edge, and the second example is a wing tip grid (Fig. 24). The point that is being made here is that a variety of techniques and software tools are available for creating grids about complex configurations. It is likely that some of these tools will be developed into user friendly software systems that are highly efficient.

Conclusions

A boundary-fitted wing-fuselage grid with several grid clusterings has been produced using Coons' patch-surface definition, plane-patch intersections, and algebraic interpolation. There are several conclusions that can be drawn from this demonstration. They are:

1. It is feasible to specify an aircraft surface independent of the grid generation process, and highly developed techniques and software developed for linear aerodynamics, and model making can be used.

2. A building-block concept where several adjoining or partially overlapping physical grids map to a computational cube simplifies the three-dimensional grid generation process.

3. The intersection of a simple surface (plane) with the configuration surface establishes curves along which grid points are distributed.

4. Algebraic grid generation techniques are readily applied to obtain the boundary and interior grid of each building block once the grid on the configuration surface is defined.

References

1. Thompson, J. F.; Warsi, Z. U. A.; and Mastin, C. W.: Boundary-Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations-A Review. Journal of Computational Physics, Vol. 47, 1982, pp. 1-108.

2. Thompson, J. F.: A Survey of Grid Generation Techniques in Computational Fluid Dynamics. AIAA Paper 83-0447, January 1983.

3. Smith, R. E.: Two-Boundary Grid Generation for the Solution of the Three-Dimensional Navier-Stokes Equations. NASA TM-83123, May 1981.

4. Smith, R. E.; and Weigle, B. L.: Analytical and Approximate Boundary-Fitted Curvilinear Coordinate Systems for Fluid Flow Simulation. AIAA Paper 80-0192, January 1980.

5. Coons, S. A.: Surfaces for Computer Aided Design of Space Forms. MAC-TR-41 (Contract No. AF-336600-42859, Massachusetts Institute of Technology), June 1967.

6. Smith, R. E.: Algebraic Grid Generation. Numerical Grid Generation. Elsevier Sciences Publishing Co., 1982, pp. 137-168.

7 Smith, R. E.; Kudlinski, R. A.; and Pitts, J. I.: Surface Grid Generation for Wing-Fuselage Bodies. NASA CP 2272, April 1983.

8 Smith, R. E.; Kudlinski, R. A.; and Everton, E. A.: Algebraic Grid Generation for Wing-Fuselage Bodies. AIAA Paper84-0002.

9. Craidon, C. B.: Description of a Digital Computer Progran for Airplane Configuration Plots. NASA TM X-2047, 1970.

10. Craidon, C. B.: A Computer Program for Fitting Smooth Surfaces to an Aircraft Configuration and Other Three-Dimensional Geometries. NASA TM X-3206, June 1975.

11. Smith, R. E.; Kudlinski, R. A.; and Everton, E. L.: A Grid Spacing Control Technique for Algebraic Grid Generation Methods. AIAA Paper 82-0226, January 1982.

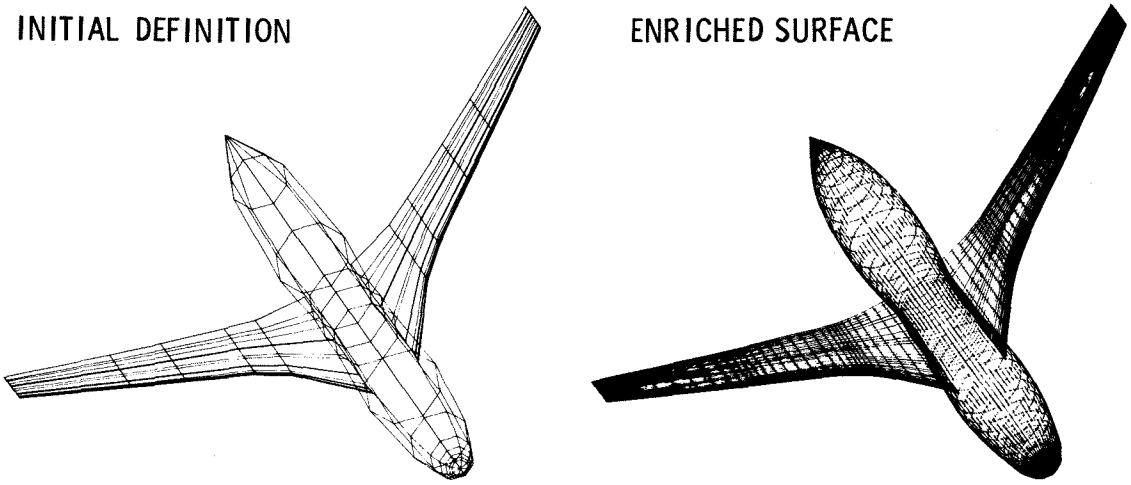

Fig. 1 Wing-Fuselage surface definition.

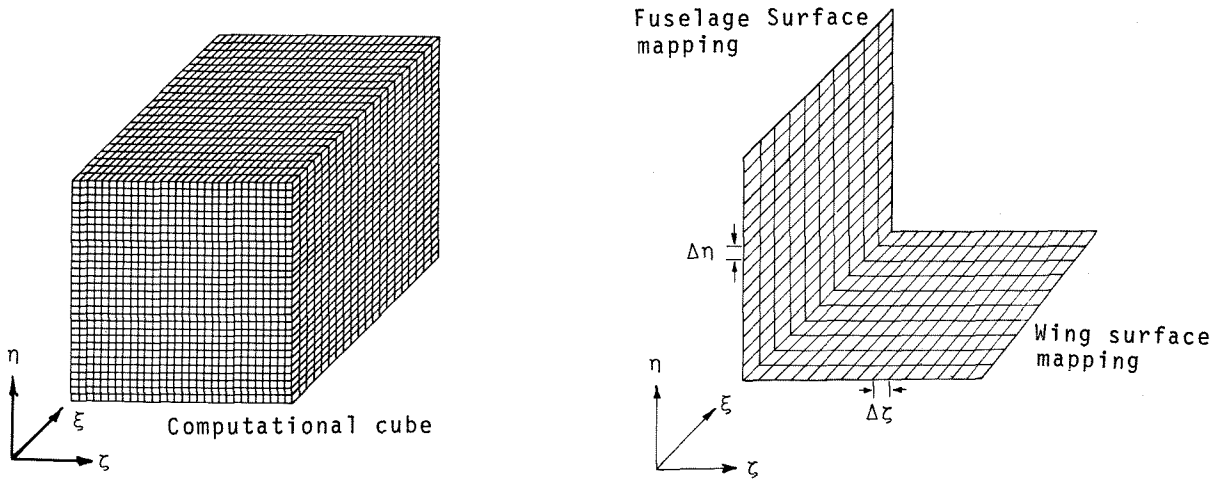

Fig. 2 Mapping to a computational domain.

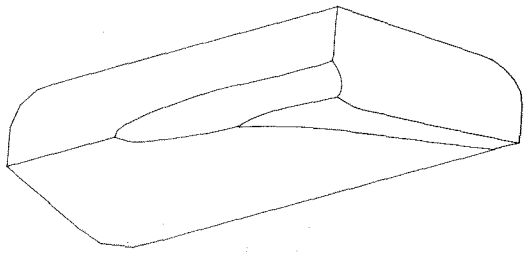

Fig. 3 Physical boundaries corresponding to the edges of the computational cube.

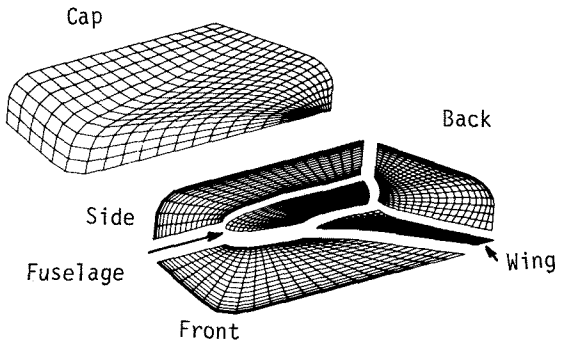

Fig. 4 Physical grid surfaces corresponding to the sides of the computational cube.

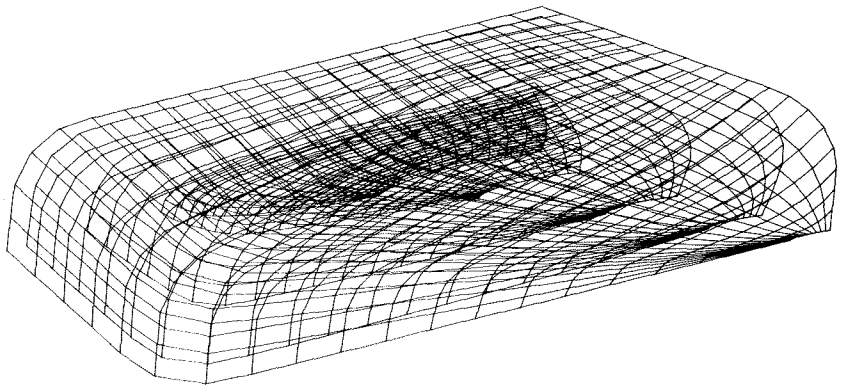

Fig. 5 Interior physical grid surfaces.

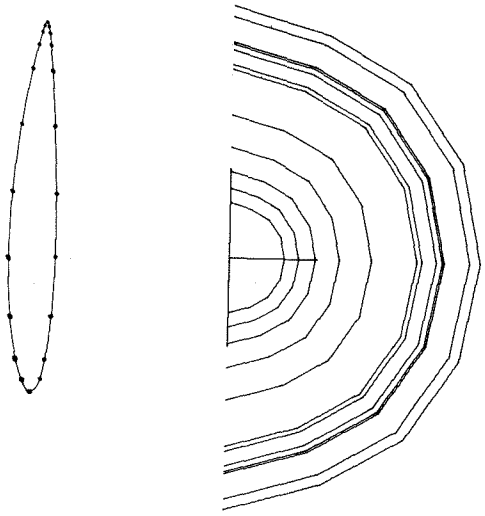

Fig. 6 Cross section definition of the wing and fuselage.

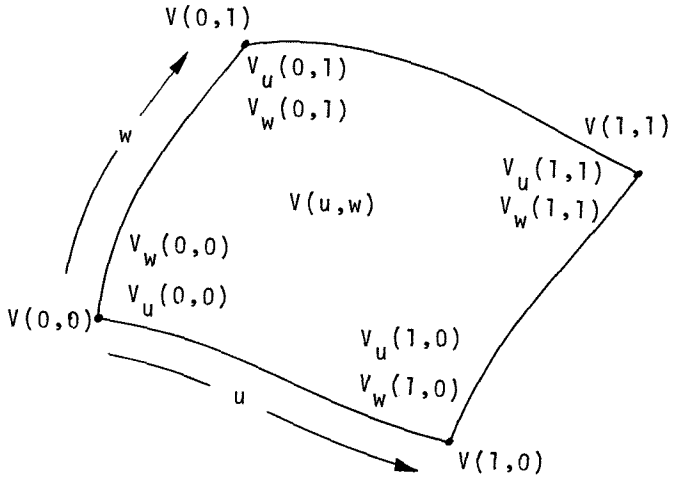

Fig. 7 Coons' patch description.

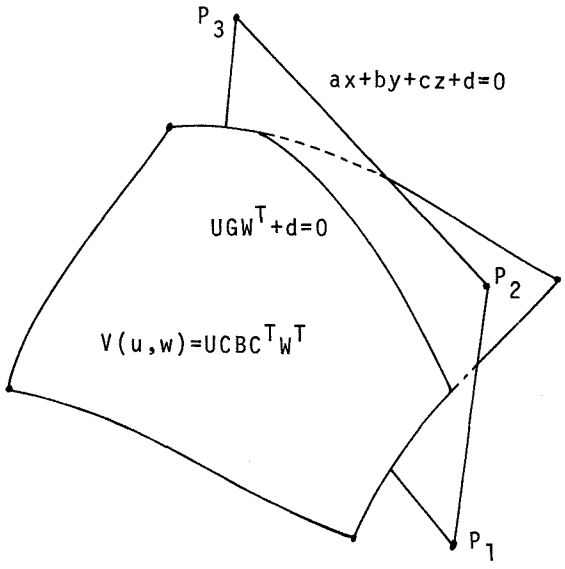

Fig. 8 Patch-plane intersection.

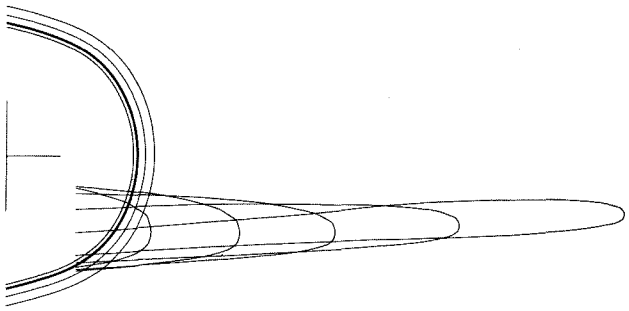

Fig. 9 Intersection of planes orthogonal to the x-axis with the wing and fuselage.

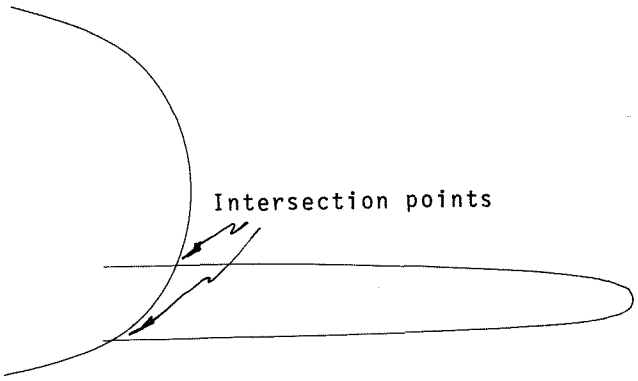

Fig. 10 Wing-fuselage intersection.

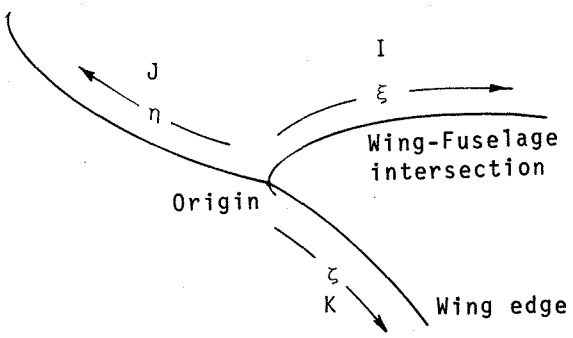

Fig. 11 Coordinate axes.

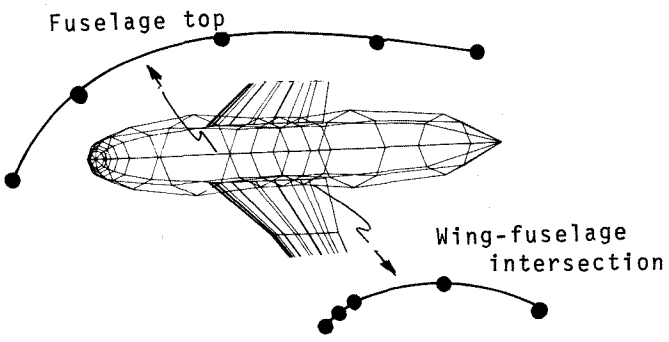

Fig. 12 Boundary grid curves for fuselage grid definition.

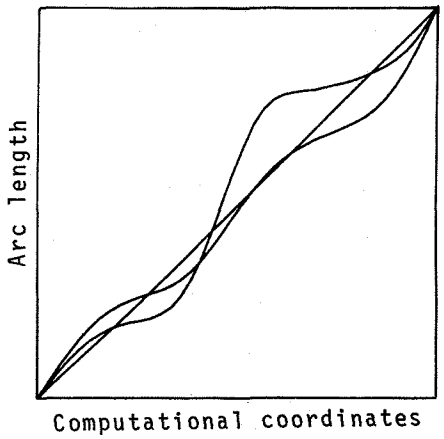

Fig. 13 Grid spacing control.

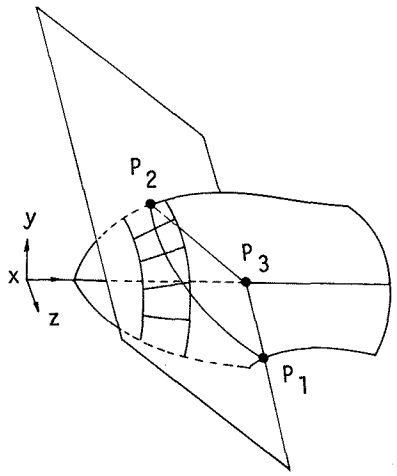

Fig. 14 Fuselage-plane intersection.

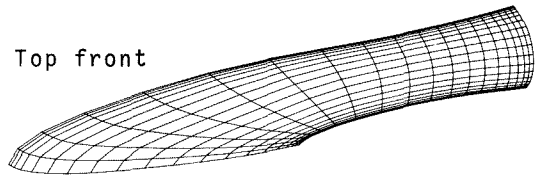

Fig. 15 Fuselage grid.

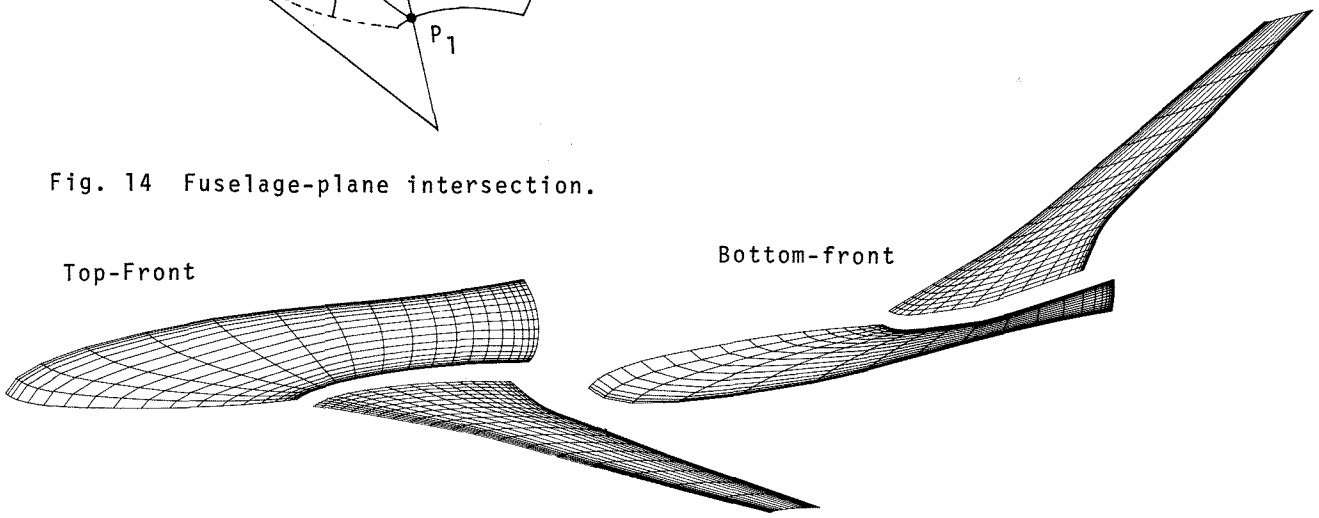

Fig. 16 Wing and fuselage grid.

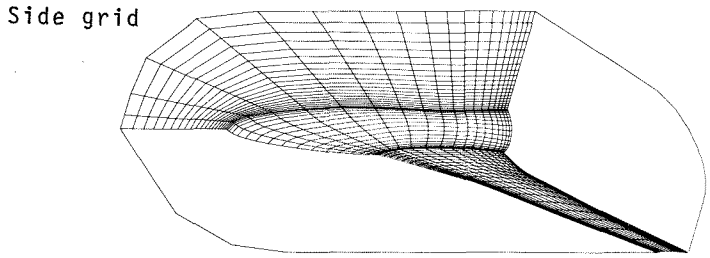

Fig. 17 Side grid.

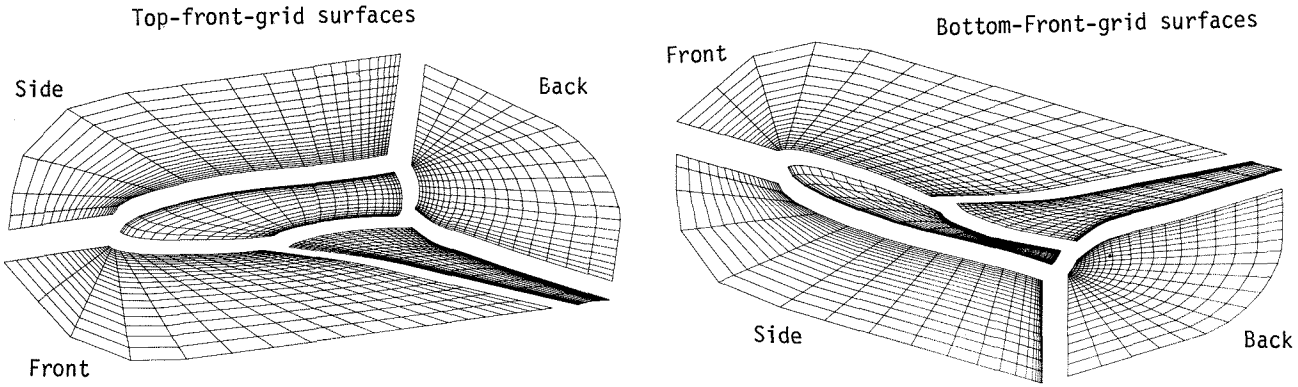

Fig. 18 Side-Front-Back grids.

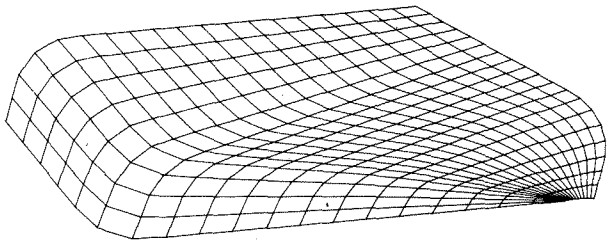

Fig. 19 Cap grid.

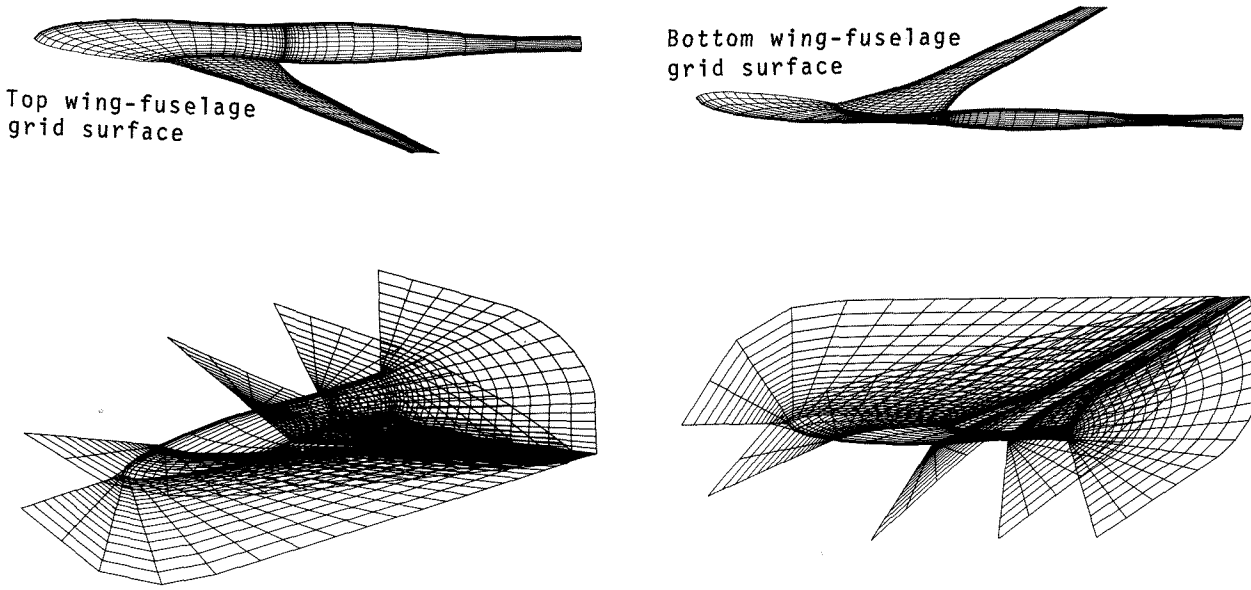

Fig. 20 Selected interior grid surfaces for the top-front and bottom-front grids.

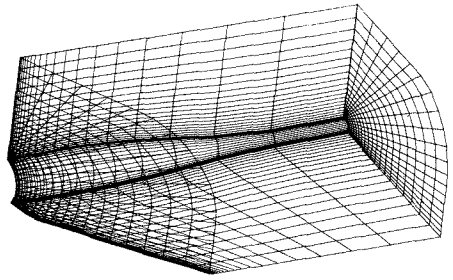

Fig. 21 Boundary grid surfaces for top-back grid.

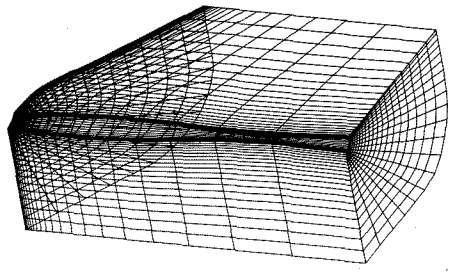

Fig.22 Boundary grid surfaces for bottom-back grid.

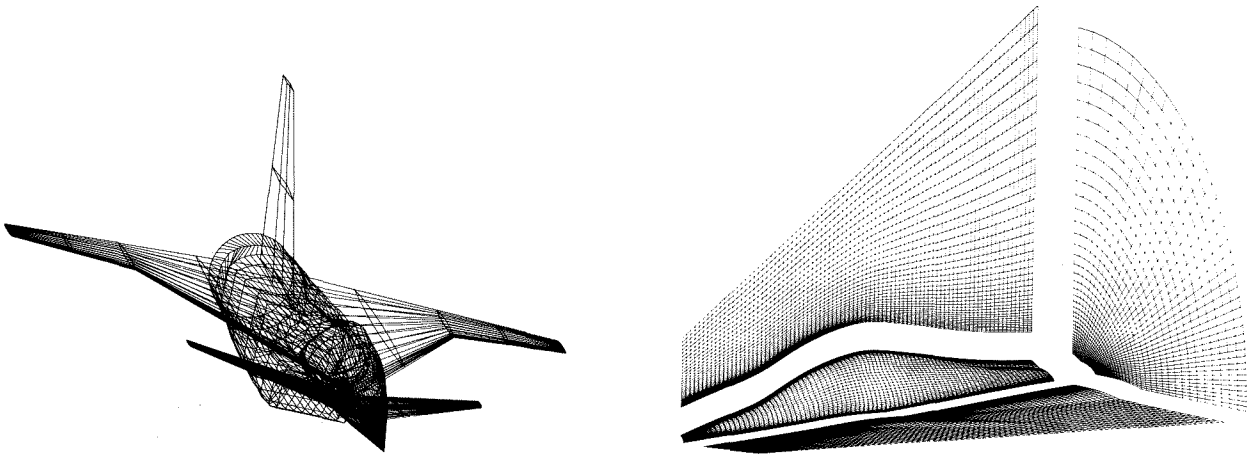

Fig. 23 Grids for a complex fighter configuration.

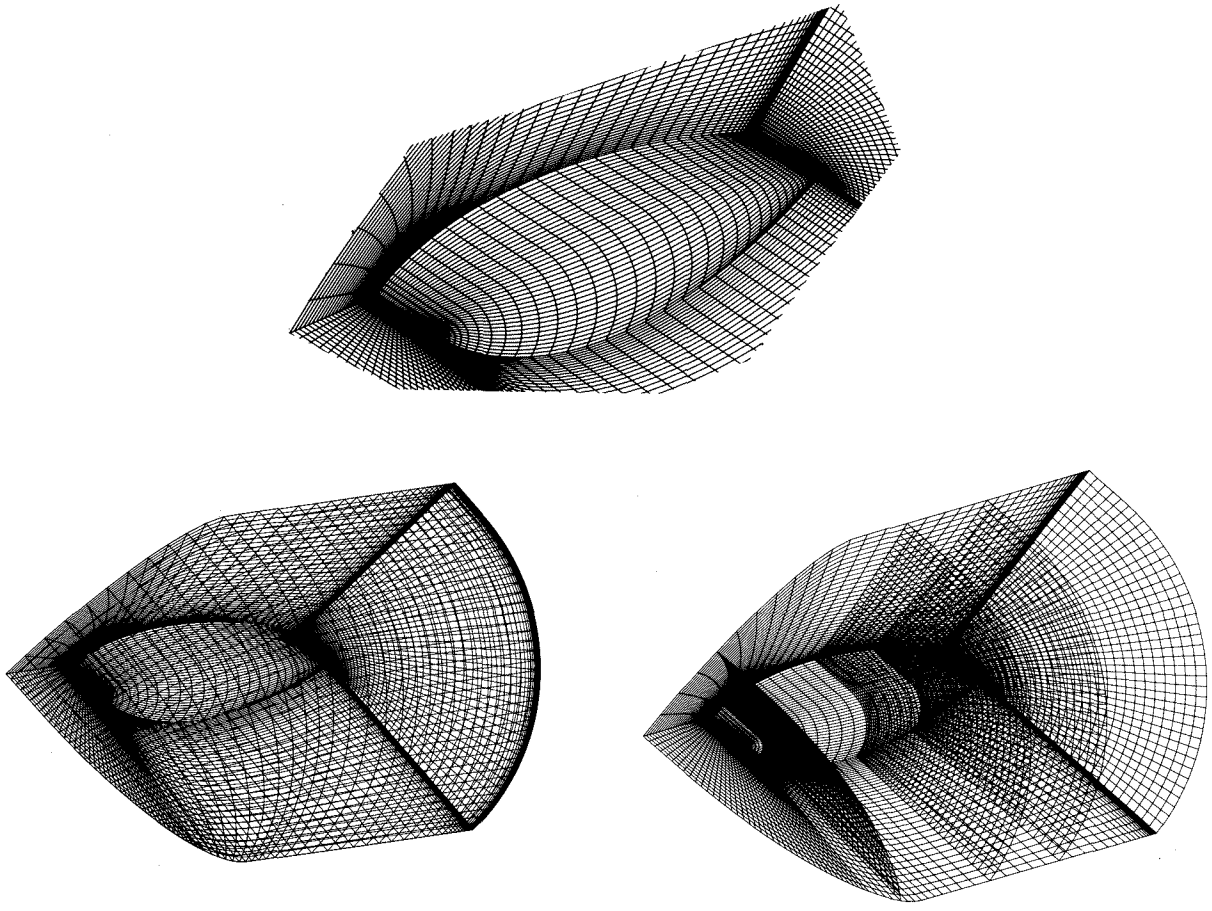

Fig. 24 Wing tip grid